IN THE CLAIMS

Please amend the claims as follows.

1. (Currently Amended) A MPEG decoder, comprising: ~~having~~

a controller capable of detecting ~~that detects~~ start codes in bitstreams received in said MPEG decoder, each of said start codes having a three-byte start code prefix and a one-byte start code value, said controller operable to:

~~(i)~~ fetch a thirty-two bit word of a received bitstream,

~~(ii)~~ determine whether a start code prefix and a start code value are properly aligned within said thirty-two bit word, and

~~(iii)~~ if not properly aligned within said thirty-two bit word, determine ~~whether the~~ that the thirty-two bit word does not contain any portion of the start code prefix based solely on a determination that a least significant byte of said thirty-two bit word ~~may be~~ is not part of said start code prefix.

2. (Currently Amended) The MPEG decoder as set forth in Claim 1 wherein said controller is further operable, if the least significant byte is not part of said start code prefix, to fetch another thirty-two bit word of said received bitstream.

3. (Currently Amended) The MPEG decoder as set forth in Claim [[2]] 1 wherein said controller is further operable to ~~(iv)~~ determine whether said start code prefix is within the three least significant bytes of said thirty-two bit word.

4. (Currently Amended) The MPEG decoder as set forth in Claim [[2]] 1 wherein said controller is further operable to ~~(iv)~~ determine whether part of said start code prefix may be within the most significant byte of a next thirty-two bit word.

5. (Original) The MPEG decoder as set forth in Claim 4 wherein said controller is further operable to fetch said next thirty-two bit word of said received bitstream.

6. (Currently Amended) The MPEG decoder as set forth in Claim 4 wherein said controller is further operable to ~~(v)~~ determine whether part of said start code prefix is within the two least significant bytes of said thirty-two bit word and the most significant byte of said next thirty-two bit word.

7. (Currently Amended) The MPEG decoder as set forth in Claim 4 wherein said controller is further operable to ~~(v)~~ determine whether part of said start code prefix is within the least significant byte of said thirty-two bit word and the two most significant bytes of said next thirty-two bit word.

8. (Currently Amended) A digital video recorder capable of playing back a recorded program stream, said digital video recorder comprising:

a video processor capable of receiving an incoming program stream and converting said incoming program stream to a baseband signal capable of being displayed on a television associated with said digital video recorder;

a storage disk capable of storing program streams for time-shifted viewing; and

a MPEG decoder capable of decoding received bitstreams and generating PES packets, said MPEG decoder comprising ~~having~~ a controller capable of detecting ~~that detects~~ start codes in said received bitstreams, each of said start codes having a three-byte start code prefix and a one-byte start code value, said controller operable to:

~~(i)~~ fetch a thirty-two bit word of a received bitstream,

~~(ii)~~ determine whether a start code prefix and a start code value are properly aligned within said thirty-two bit word, and

~~(iii)~~ if not properly aligned within said thirty-two bit word, determine ~~whether the~~ that the thirty-two bit word does not contain any portion of the start code prefix based solely on a determination that a least significant byte of said thirty-two bit word ~~may be~~ is not part of said start code prefix.

9. (Currently Amended) The digital video recorder as set forth in Claim 8 wherein said controller is further operable, if the least significant byte is not part of said start code prefix, to fetch another thirty-two bit word of said ~~recorded~~ received bitstream.

10. (Currently Amended) The digital video recorder as set forth in Claim [[9]] 8 wherein said controller is further operable to ~~(iv)~~ determine whether said start code prefix is within the three least significant bytes of said thirty-two bit word.

11. (Currently Amended) The digital video recorder as set forth in Claim [[9]] 8 wherein said controller is further operable to ~~(iv)~~ determine whether part of said start code prefix may be within the most significant byte of a next thirty-two bit word.

12. (Original) The digital video recorder as set forth in Claim 11 wherein said controller is further operable to fetch said next thirty-two bit word of said received bitstream.

13. (Currently Amended) The digital video recorder as set forth in Claim 11 wherein said controller is further operable to ~~(v)~~ determine whether part of said start code prefix is within the two least significant bytes of said thirty-two bit word and the most significant byte of said next thirty-two bit word.

14. (Currently Amended) The digital video recorder as set forth in Claim 11 wherein said controller is further operable to ~~(v)~~ determine whether part of said start code prefix is within the least significant byte of said thirty-two bit word and the two most significant bytes of said next thirty-two bit word.

15. (Currently Amended) A method of detecting start codes in bitstreams received in a MPEG decoder, each of said start codes having a three-byte start code prefix and a one-byte start code value, said method comprising the steps of:

~~(i)~~ fetching a thirty-two bit word of a received bitstream;

~~(ii)~~ determining whether a start code prefix and a start code value are properly aligned within said thirty-two bit word; and

~~(iii)~~ if not properly aligned within said thirty-two bit word, determining ~~whether the~~ that the thirty-two bit word does not contain any portion of the start code prefix based solely on a determination that a least significant byte of said thirty-two bit word ~~may be~~ is not part of said start code prefix.

16. (Currently Amended) The method as set forth in Claim 15 further comprising the step of ~~(iv)~~ determining whether said start code prefix is within the three least significant bytes of said thirty-two bit word.

17. (Currently Amended) The method as set forth in Claim 15 further comprising the step of ~~(iv)~~ determining whether part of said start code prefix may be within the most significant byte of a next thirty-two bit word.

18. (Original) The method as set forth in Claim 17 further comprising the step of fetching said next thirty-two bit word of said received bitstream.

19. (Currently Amended) The method as set forth in Claim 15 further comprising the step of ~~(v)~~ determining whether part of said start code prefix is within the two least significant bytes of said thirty-two bit word and the most significant byte of a next thirty-two bit word.

20. (Currently Amended) The method as set forth in Claim 15 further comprising the step of ~~(v)~~ determining whether part of said start code prefix is within the least significant byte of said thirty-two bit word and the two most significant bytes of a next thirty-two bit word.